

Name: Isabel Ogilvie
Database Final Project Part 3

a. Develop SQL code to create the entire database schema, reflecting the constraints identified in previous steps.

```
# passing queries to cursor by storing query in string variable
staff_query = """
CREATE TABLE IF NOT EXISTS Staff (
    staffNo VARCHAR(8) NOT NULL PRIMARY KEY,
    name VARCHAR(35) NOT NULL,
    phoneNo INT NOT NULL UNIQUE,
    address VARCHAR(50) NOT NULL,
    DOB TEXT,
    position VARCHAR(25) NOT NULL,
    salary INT,
    clinicNo VARCHAR(8) NOT NULL,
    FOREIGN KEY (clinicNo) REFERENCES Clinic
);
"""

clinic_query = """
CREATE TABLE IF NOT EXISTS Clinic (
    clinicNo VARCHAR(8) NOT NULL PRIMARY KEY,
    name VARCHAR(35),
    address VARCHAR(50) UNIQUE,
    phoneNo INT NOT NULL UNIQUE,
    managerNo VARCHAR(8) NOT NULL,
    FOREIGN KEY (managerNo) REFERENCES Staff(staffNo)
);
"""
```

```

pet_query = """
    CREATE TABLE IF NOT EXISTS Pet (
        petNo VARCHAR(8) NOT NULL PRIMARY KEY,
        name VARCHAR(35) NOT NULL,
        species VARCHAR(25) NOT NULL,
        breed VARCHAR(25) NOT NULL,
        color VARCHAR(15),
        DOB TEXT,
        clinicNo VARCHAR(8) NOT NULL,
        ownerNo VARCHAR(8) NOT NULL,
        FOREIGN KEY (ownerNo) REFERENCES Owner,
        FOREIGN KEY (clinicNo) REFERENCES Clinic
    );
    """

```

```

pet_owner_query = """
    CREATE TABLE IF NOT EXISTS PetOwner(
        ownerNo VARCHAR(8) NOT NULL PRIMARY KEY,
        name VARCHAR(35),
        phoneNo INT,
        address VARCHAR(50) NOT NULL
    );
    """

```

```

examination_query = """
    CREATE TABLE IF NOT EXISTS Examination (
        examNo VARCHAR(8) NOT NULL PRIMARY KEY,
        chiefComplaint VARCHAR(500) NOT NULL,
        description VARCHAR(500),
        dateSeen TEXT NOT NULL,
        actions VARCHAR(100),
        petNo VARCHAR(10) NOT NULL,
        staffNo VARCHAR(10) NOT NULL,
        FOREIGN KEY (petNo) REFERENCES Pet,
        FOREIGN KEY (staffNo) REFERENCES Staff
    );
    """

```

```

cursor.execute(staff_query)
cursor.execute(clinic_query)
cursor.execute(pet_query)
cursor.execute(pet_owner_query)
cursor.execute(examination_query)

```

b. Create at least 5 tuples for each relation in your database.

```
# Create at Least 5 tuples for each relation in your database
```

```
insert_clinic_rows = """
```

```
    INSERT OR IGNORE INTO Clinic
```

```
    VALUES
```

```
        ('CL001', 'Paws and Claws Westbrook Clinic', '123 Canine St', 7544654672, 'ST045'),
        ('CL002', 'Paws and Claws Knight Clinic', '456 Feline Rd', 7891234567, 'ST023'),
        ('CL003', 'Miami-Dade Regional Clininc', '789 Reptile Ave', 2649876543, 'ST009'),
        ('CL004', 'Central Florida Clinic ', '777 Bird Lane', 7542018786, 'ST001'),
        ('CL005', 'Kingston Downtown Pet Clinic', '101 Fish Blvd', 7542098735, 'ST076');
```

```
"""
```

```
insert_staff_rows = """
```

```
    INSERT OR IGNORE INTO Staff
```

```
    VALUES
```

```
        ('ST045', 'Jane Doe', 1234567890, '267 Gingerbread Lane', '1989-12-25', 'Manager', 40000, 'CL001'),
        ('ST023', 'Jack Doe', 7544567899, '321 Mystery Blvd', '1986-09-01', 'Manager', 40000, 'CL002'),
        ('ST009', 'Ben Richards', 7544678902, '425 London Rd', '1994-06-28', 'Manager', 45000, 'CL003'),
        ('ST001', 'Diego Gogo', 7548791234, '117 Swiper St', '1979-02-15', 'Manager', 40000, 'CL004'),
        ('ST076', 'Dora Explorer', 7890001111, '123 Boots Ave', '1990-11-10', 'Manager', 50000, 'CL005');
```

```
"""
```

```
insert_pet_rows = """
```

```
    INSERT OR IGNORE INTO Pet
```

```
    VALUES
```

```
        ('P0001', 'Simba', 'Cat', 'Maine Coon', 'Grey', '2020-09-11', 'OW001', 'CL001'),
        ('P0002', 'Sagikor', 'Cat', 'Ragdoll', 'White', '2016-07-18', 'OW003', 'CL002'),
        ('P0003', 'Hershey', 'Dog', 'German Sheperd', 'Brown', '2019-01-01', 'OW002', 'CL003'),
        ('P0004', 'Bobby', 'Dog', 'Akita', 'Black', '2013-04-30', 'OW004', 'CL004'),
        ('P0005', 'Sunshine', 'Bird', 'Parrot', 'Green', '2018-12-14', 'OW005', 'CL005');
```

```
"""
```

```
insert_pet_owner_rows = """
```

```
    INSERT OR IGNORE INTO PetOwner
```

```
    VALUES
```

```
        ('OW001', 'James Madison', 7453627899, '123 Miracle Dr'),
        ('OW002', 'Isabel Kathryn', 7542098716, '456 Song Blvd'),
        ('OW003', 'Joseph McFarlane', 7542098777, '789 Disney Ave'),
        ('OW004', 'Joanna Allison', 7897543672, '400 Miami Lane'),
        ('OW005', 'Peter Johnson', 7890002678, '3756 Jamaica St');
```

```
"""
```

```
insert_examination_rows = """
```

```
    INSERT OR IGNORE INTO Examination
```

```
    VALUES
```

```
        ('EX001', 'Monthly Physical Check-Up', 'Monthly physical well being check', '2022-11-21', 'None', 'P0001', 'ST001'),
        ('EX002', 'Dental Cleaning', 'Special cleaning to treat gingivitis', '2022-11-22', 'Full cleaning done', 'P0002', 'ST076'),
        ('EX003', 'Vaccination', 'Got tetanus shot', '2022-11-23', 'Vaccination administered', 'P0003', 'ST009'),
        ('EX004', 'Neutering', 'Had tubes tied', '2022-11-24', 'Wound Cleaning solution prescribed', 'P0004', 'ST023'),
        ('EX005', 'Monthly Physical Check-Up', 'Monthly physical well being check', '2022-12-01', 'None', 'P0005', 'ST045');
```

```
"""
```

```
cursor.execute(insert_staff_rows)
```

```
cursor.execute(insert_clinic_rows)
```

```
cursor.execute(insert_pet_rows)
```

```
cursor.execute(insert_pet_owner_rows)
```

```
cursor.execute(insert_examination_rows)
```

=====								
	clinicNo		name		address	managerNo	phoneNo	
0	CL001	Paws and Claws	Westbrook Clinic	123	Canine St	7544654672	ST045	
1	CL002	Paws and Claws	Knight Clinic	456	Feline Rd	7891234567	ST023	
2	CL003	Miami-Dade	Regional Clininc	789	Reptile Ave	2649876543	ST009	
3	CL004		Central Florida Clinic	777	Bird Lane	7542018786	ST001	
4	CL005	Kingston	Downtown Pet Clinic	101	Fish Blvd	7542098735	ST076	
=====								
	staffNo		name		address	phoneNo	DOB	position
0	ST045	Jane Doe	1234567890	267	Gingerbread Lane	1989-12-25	Manager	40000
1	ST023	Jack Doe	7544567899	321	Mystery Blvd	1986-09-01	Manager	40000
2	ST009	Ben Richards	7544678902	425	London Rd	1994-06-28	Manager	45000
3	ST001	Diego Gogo	7548791234	117	Swiper St	1979-02-15	Manager	40000
4	ST076	Dora Explorer	7890001111	123	Boots Ave	1990-11-10	Manager	50000
=====								
	petNo		name	species	breed	DOB	color	ownerNo
0	P0001	Simba	Cat	Maine Coon	Grey	2020-09-11	OW001	CL001
1	P0002	Sagikor	Cat		Ragdoll	White	2016-07-18	OW003
2	P0003	Hershey	Dog	German Sheperd	Brown	2019-01-01	OW002	CL003
3	P0004	Bobby	Dog		Akita	Black	2013-04-30	OW004
4	P0005	Sunshine	Bird		Parrot	Green	2018-12-14	OW005
=====								
	ownerNo		name		phoneNo		address	
0	OW001	James Madison	7453627899	123	Miracle Dr			
1	OW002	Isabel Kathryn	7542098716	456	Song Blvd			
2	OW003	Joseph McFarlane	7542098777	789	Disney Ave			
3	OW004	Joanna Allison	7897543672	400	Miami Lane			
4	OW005	Peter Johnson	7890002678	3756	Jamaica St			
=====								
	examNo		chiefComplaint		description		dateSeen	
0	EX001	Monthly Physical Check-Up	Monthly physical well being check				2022-11-21	
1	EX002	Dental Cleaning	Special cleaning to treat gingivitis				2022-11-22	Full cle
2	EX003	Vaccination			Got tetanus shot		2022-11-23	Vaccination ad
3	EX004	Neutering			Had tubes tied		2022-11-24	Wound Cleaning solution
4	EX005	Monthly Physical Check-Up	Monthly physical well being check				2022-12-01	

c. Develop 5 SQL queries using embedded SQL (see Python tutorial).

```

clinic_query = """
SELECT * FROM Clinic
"""
staff_query = """
SELECT * FROM Staff
"""
pet_query = """
SELECT * FROM Pet
"""
pet_owner_query = """
SELECT * FROM PetOwner
"""
examination_query = """
SELECT * FROM Examination
"""

querylist = [clinic_query, staff_query, pet_query, pet_owner_query, examination_query]

print("=====")
for q in querylist:
    cursor.execute(q)
    column_names = [row[0] for row in cursor.description]
    table_data = cursor.fetchall()
    df = pd.DataFrame(table_data, columns=column_names)
    print("=====")
    print(df)

instruction1 = "List the pet names, and pet's owner names, breeds and species for all pets whose owner's name starts with a J"
query1 = """
SELECT p.name AS pet_name, o.name AS owner_name, p.species, p.breed
FROM Pet p, PetOwner o
WHERE p.ownerNo = o.ownerNo AND o.name LIKE 'J%'
"""
instruction3 = "List all the managers "

instruction4 = "List all examinations number, complaint and date seen and petNo"
query4 = """
SELECT examNo, dateSeen, chiefComplaint, petNo
FROM Examination
"""
instruction5 = "List the amount of staff working at each clinic"
query5 = """
SELECT c.clinicNo, COUNT(staffNo) AS staff_amt
FROM Clinic c, Staff s
WHERE s.clinicNo = c.clinicNo
GROUP BY c.clinicNo
"""
instruction2 = "List all pet owner's pets."
query2 = """
SELECT petNo, p.ownerNo, o.name
FROM PetOwner o, Pet p
WHERE o.ownerNo = p.ownerNo
"""

queries = [(query1, instruction1), (query2, instruction2), (query3, instruction3), (query4, instruction4), (query5, instruction5)]
print("*****")
print("QUERIES")

for (query, instruction) in queries:
    cursor.execute(query)
    column_names = [row[0] for row in cursor.description]
    table_data = cursor.fetchall()
    df = pd.DataFrame(table_data, columns=column_names)
    print("*****")
    print(instruction)
    print(df)
print("*****")

db_connect.close()

```

```

*****
QUERIES
*****
List the pet names, and pet's owner names, breeds and species for all pets whose owner's name starts with a J
pet_name      owner_name species      breed
0  Simba      James Madison   Cat      Maine Coon
1  Sagicor    Joseph McFarlane Cat      Ragdoll
2  Bobby      Joanna Allison   Dog      Akita
*****

List all pet owner's pets.
petNo ownerNo      name
0  P0001  OW001      James Madison
1  P0002  OW003      Joseph McFarlane
2  P0003  OW002      Isabel Kathryn
3  P0004  OW004      Joanna Allison
4  P0005  OW005      Peter Johnson
*****

List all the managers
staffNo      name position      salary
0  ST045      Jane Doe   Manager      40000
1  ST023      Jack Doe   Manager      40000
2  ST009      Ben Richards Manager      45000
*****

List all examinations number, complaint and date seen and petNo
examNo      dateSeen      chiefComplaint petNo
0  EX001  2022-11-21  Monthly Physical Check-Up ST001
1  EX002  2022-11-22      Dental Cleaning ST076
2  EX003  2022-11-23      Vaccination ST009
3  EX004  2022-11-24      Neutering ST023
4  EX005  2022-12-01  Monthly Physical Check-Up ST045
*****

*****

List the amount of staff working at each clinic
clinicNo      staff_amt
0  CL001      1
1  CL002      1
2  CL003      1
3  CL004      1
4  CL005      1
*****

```

d. Upload all the code and documentation to GitHub.

See github repository at : <https://github.com/isabel-png/csc423>